university of groningen

Advanced metric adaptation in LVQ

Petra Schneider   Michael Biehl   Kerstin Bunte

Institute of Mathematics and Computing Science, University of Groningen, the Netherlands
{p.schneider, m.biehl, k.bunte}@rug.nl

Learning Vector Quantization

Learning Vector Quantization (LVQ) is an intuitive and powerful family of prototype-based classification algorithms. Successful applications of LVQ include such diverse problems like medical image or data analysis fault detection in technical systems, or the classification of satellite spectral data. Consider a finite set of training data $\{(\xi_k, y_k)\}_{k=1}^{K}$ given, where $\xi \in \mathbb{R}^n$ and $y \in \{1, \ldots, C\}$ specifies the class label sample ξ. From this data, LVQ algorithms learn of a set of labeled prototypes $W = \{w_i, c(w_i)|\mathbb{R}^n \times \{1, \ldots, C\}\}_{i=1}^{l}$, which approximate the classes. After training, the prototypes are used for a nearest prototype classification. The performance of LVQ systems highly depends on the use of an appropriate distance measure which has to represent the underlying structure of the data appropriately.

Cost function based training

The first LVQ training algorithms are based on heuristics. Later, algorithms which optimize a cost function were proposed. In our work, we focus on two different approaches:

- Generalized LVQ (GLVQ) [2]: Based on a heuristic cost function which aims at margin optimization

$$E_{GLVQ} = \sum_k \phi\left(\frac{d(\xi_k, w_J) - d(\xi_k, w_K)}{d(\xi_k, w_J) + d(\xi_k, w_K)}\right)$$

  w_J: closest prototype with $c(w) = y_k$,
  w_K: closest prototype with $c(w) \neq y_k$,
  ϕ: monotonically increasing function, e.g. identity, tanh.

- Robust Soft LVQ (RSLVQ) [3]: Based on a statistical modeling of the given data distribution, i.e. the probability density is described by a mixture modeling. The algorithm maximizes the likelihood ratio

$$L = \prod_k L(\xi_k, y_k), \text{ where } \quad L(\xi_k, y_k) = \frac{p(\xi_k, y_k|W)}{p(\xi_k|W)}$$

  $p(\xi, y|W) = \sum_{j:c(w_j)=y} p(\xi|j)P(j)$,
  $p(\xi|W) = \sum_j p(\xi|j)P(j)$,
  $p(\xi|j) = K(j) \cdot \exp f(\xi, w_j, \sigma_j^2)$.
  The RSLVQ cost function is defined in terms of $E_{RSLVQ} = \log(L)$.

Both methods optimize E based on the gradient information.

Metric adaptation

Parameterized distance measures allow to optimize the metric for every given data distribution individually.

- Relevance learning: Extend the Euclidean distance by an adaptive weight vector $\lambda \in \mathbb{R}^n, \lambda_i > 0, |\lambda| = 1$

$$d^{\lambda}(\xi, w) = \sum_{i=1}^{n} \lambda_i (\xi_i - w_i)^2.$$

After training, λ reflects the importance of the input features for classification. The approach allows to eliminate irrelevant/noisy dimension.

Relevance Matrices in LVQ

We propose to extend the squared Euclidean distance by a full matrix $\Lambda \in \mathbb{R}^{n \times n}$ of adaptive weight values

$$d^{\Lambda}(\xi, w) = (\xi - w)^T \Lambda (\xi - w),$$

where $\sum_i \Lambda_{ii} = 1$. We substitute $\Lambda = \Omega^T \Omega$, $\Omega \in \mathbb{R}^{m \times n}, m \leq n$ to guarantee positive definiteness.

- Generalization of relevance learning, $\mathrm{diag}(\Lambda) \equiv \lambda$
- Additionally takes correlations into account
- Eigenvectors define new feature set
- $d^{\Lambda}(\xi, w) = [\Omega(\xi - w)]^2$
- Localized extension: $d^{\Lambda_j}(\xi, w_j) = (\xi - w_j)^T \Lambda_j (\xi - w_j)$

For training, the squared Euclidean distance in E_{GLVQ} and E_{RSLVQ} is replaced by d^{Λ}. The new cost functions are optimized with respect to the prototypes and Ω_{lm}. We name the new algorithms Generalized Matrix LVQ (GMLVQ) [1] and Matrix Robust Soft LVQ (MRSLVQ) [1].

Results

Application: Image Segmentation Data Set (UCI Repository)
16 features, 7 classes

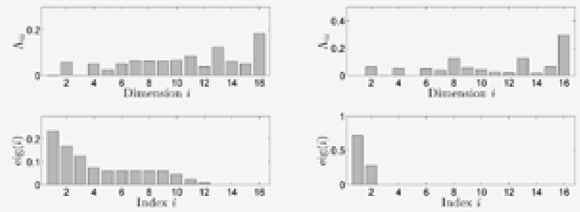

Figure 1: *Visualization of diagonal elements and eigenvalues of Λ with $\Omega \in \mathbb{R}^{16\times16}$ (left) and $\Omega \in \mathbb{R}^{2\times16}$ (right).*

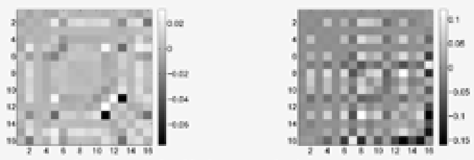

Figure 2: *Visualization of off-diagonal elements of Λ with $\Omega \in \mathbb{R}^{16\times16}$ (left) and $\Omega \in \mathbb{R}^{2\times16}$ (right).*

Classification performance:

Algorithm	ϵ_{test}
GLVQ	17%
GMLVQ($\Omega \in \mathbb{R}^{2\times16}$)	16.5%
GMLVQ($\Omega \in \mathbb{R}^{16\times16}$)	9.8%
Local GMLVQ	5.6%

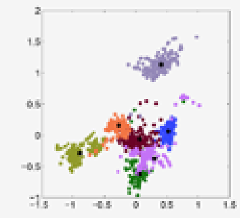

Figure 3: *Data transformed by $\Omega \in \mathbb{R}^{2\times16}$.*

Regularization in Matrix relevance learning

- Aim: Prevent metric adaptation techniques from eliminating too many dimensions
- Proposed regularization term: $\theta = \ln(\det(\Omega\Omega^T))$
- θ punishes strong decays in the eigenvalue profile of Λ
- Extended GMLVQ cost function: $\tilde{E}_{GMLVQ} = E_{GMLVQ} - \frac{\eta}{2} \cdot \theta$

Application: Pima indians diabetes data (UCI Repository)
8 features, 2 classes

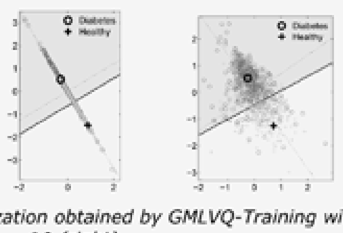

Figure 4: *Visualization obtained by GMLVQ-Training with $\Omega \in \mathbb{R}^{2\times8}$ and $\eta = 0$ (left), $\eta = 0.2$ (right).*

Acknowledgement

This work was supported by the "Nederlandse organisatie voor Wetenschappelijke Onderzoek (NWO)" (project 612.066.620).

References

[1] P. Schneider, M. Biehl, B. Hammer, Distance learning in discriminative vector quantization, Neural Computation 21 (10) (2009) 2942--2969.

[2] A. Sato, K. Yamada, Generalized Learning Vector Quantization, in: M. C. M. D. S. Touretzky, M. E. Hasselmo (Eds.), Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference, MIT Press, Cambridge, MA, USA, 1996, pp. 423--9.

[3] S. Seo, K. Obermayer, Soft Learning Vector Quantization, Neural Computation 15 (7) (2003) 1589--1604.

SIREN 2009, Scientific ICT Research Event Netherlands, November 5, 2009, University of Twente, Enschede